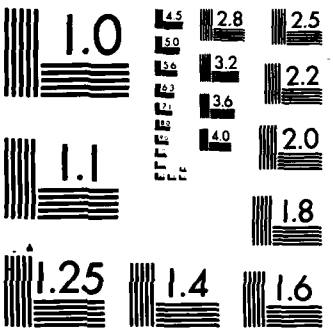
AD-A159 252 PARALLEL MATRIX COMPUTATIONS(U) MARYLAND UNIV COLLEGE PARK G W STEWART ET AL. 11 APR 85 AFOSR-TR-85-0695 AFOSR-82-0078 1/1

UNCLASSIFIED F/G 9/2 NL



Interim Report

Grant AFOSR 82-0078

9 February 1983 - 16 May 1985

Title

PARALLEL MATRIX COMPUTATIONS

Principal Investigator

G. W. Stewart

University of Maryland
College Park MD 20742

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

AD-A159 252

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY -----	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; Distribution Unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S) AFOSR-TR- 85-0695
6a. NAME OF PERFORMING ORGANIZATION University of Maryland 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION AFOSR
6c. ADDRESS (City, State and ZIP Code) College Park MD 20742	7b. ADDRESS (City, State and ZIP Code) Bldg. 410 Bolling AFB D.C. 20332-6448
8a. NAME OF FUNDING/SPONSORING ORGANIZATION AFOSR 8b. OFFICE SYMBOL (If applicable) NM	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER AFOSR-82-0078
8c. ADDRESS (City, State and ZIP Code) Bldg. 410 Bolling AFB D.C. 20332-6448	10. SOURCE OF FUNDING NOS. PROGRAM ELEMENT NO. 61102F; PROJECT NO. 2304; TASK NO. A3; WORK UNIT NO. ----
11. TITLE (Include Security Classification) Parallel Matrix Computations	
12. PERSONAL AUTHOR(S) G. W. Stewart & D. P. O'Leary	
13a. TYPE OF REPORT Interim	13b. TIME COVERED FROM 9 Feb 83 TO 16 May 85
14. DATE OF REPORT (Yr., Mo., Day) 11 April 1985	15. PAGE COUNT 6
16. SUPPLEMENTARY NOTATION	

17. COSATI CODES		
FIELD	GROUP	SUB. GR.
XXXXXXXXXXXXXXXX		

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
realistic algorithms for matrix computations, ZMOB, (parallel system)

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

The purpose of this effort is to develop realistic algorithms for matrix computations on parallel computers. It has been long observed that the usual algorithms of numerical linear algebra contain a great deal of inherent parallelism. For example, if the arithmetic operations that can be performed in parallel in Gaussian elimination are actually so executed, the time to decompose an nXn matrix is reduced from n^3 to n only recently, with the emergence of cheap, small microcomputers, has it become feasible to exploit this parallelism on anything but a trivial scale.

At the Department of Computer Science at the University of Maryland, there is under development a parallel system, called ZMOB, consisting of 256 micro-processors connected on a conveyor belt. This belt is so fast and its architecture is such that any two processors can communicate without interfering with the communications of other pairs of processors. Thus the ZMOB is an ideal tool for simulating an arbitrarily connected (cont)

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☒ DTIC USERS ☐	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL John Thomas, Jr. CAPT. USAF	22b. TELEPHONE NUMBER (Include Area Code) (202)767-5026 22c. OFFICE SYMBOL AFOSR/NM

DD FORM 1473, 83 APR EDITION OF 1 JAN 73 IS OBSOLETE.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

(#19 cont.)

network of computers.

This feature of the ZMOB is particularly useful in investigating parallel matrix algorithms. As was noted above, there is much parallelism in most current matrix algorithms. However, to exploit it, information must be moved from processor to processor. This constitutes the chief bottleneck in parallel matrix algorithms; interconnections between processors are expensive, and in a practical system one can assume only a limited amount of connectivity. The ZMOB provides a means of testing and comparing different types of interconnections, since all one has to do is not use the rich connections provided by the ZMOB conveyor belt. Thus our proposal is to use the ZMOB to design and test networks for parallel matrix computations.

X

A-1

Interim Report
on
Grant AFOSR-82-0078 16 May 85
For Period Ending ~~April 14, 1984~~
D. P. O'Leary
G. W. Stewart

1. Introduction

This is a summary of work accomplished under Grant AFOSR-82-0078. The purpose of this effort is to develop realistic algorithms for matrix computations on parallel computers. It has been long observed that the usual algorithms of numerical linear algebra contain a great deal of inherent parallelism. For example, if the arithmetic operations that can be performed in parallel in Gaussian elimination are actually so executed, the time to decompose an $n \times n$ matrix is reduced from order n^3 to n Only recently, with the emergence of cheap, small microcomputers, has it become feasible to exploit this parallelism on anything but a trivial scale.

At the Department of Computer Science at the University of Maryland, there is under development a parallel system, called the ZMOB, consisting of 256 micro-processors connected on a conveyor belt. This belt is so fast and its architecture is such that any two processors can communicate without interfering with the communications of other pairs of processors. Thus the ZMOB is an ideal tool for simulating an arbitrarily connected network of computers.

This feature of the ZMOB is particularly useful in investigating parallel matrix algorithms. As was noted above, there is much parallelism in most current matrix algorithms. However, to exploit it, information must be moved from processor to processor. This constitutes the chief bottleneck in parallel matrix algorithms; interconnections between processors are expensive, and in a practical system one can assume only a limited amount of connectivity. The ZMOB provides a means of testing and comparing different types of interconnections, since all one has to do is *not* use the rich connections provided by the ZMOB conveyor belt. Thus our proposal is to use the ZMOB to design and test networks for parallel matrix computations.

2. Progress to Date

Our research is proceeding in three stages. First, decide on a suitable way of connecting and synchronizing processors for parallel matrix computations. Second, design and build a communications system to realize this network on the ZMOB. Third, code matrix algorithms for the system, and experiment with them. In addition, we must install and test the floating-point processors which were requested as part of the initial grant period. In this section we shall take up each of these points in turn.

A greater part of our research concerns two dimensional arrays of processors, and we have made considerable progress in this area. It is highly desirable to be able to restrict the connections in such an array to lines between adjacent processors, since this is the simplest and most easily implemented of networks. We have observed that not only can many matrix algorithms be implemented on such a network, but also the processors can be synchronized by the flow of data in the network, without any need of outside control. A sketch of how such networks operate was given in the first renewal proposal. Here we just list some of the advantages of the approach.

1. The interconnections are simple and realizable.
2. Each processor can operate asynchronously.
3. The same program can be used on each processor.
4. Many matrix algorithms fit naturally into this scheme.
5. The approach provides a natural way of dealing with array overflow; i.e., the case where the size of the matrix exceeds the size of the array processors.

In order to support the network, we are building a comunications system to pass information from processor to processor. This system will be invoked from a high level programming language, and it will permit multi-processing on a single processor. This latter feature is necessary to cope with array overflow. The core of the operating system has been programmed and has been used to perform small matrix computations on the ZMOB. We shall begin testing code previously written for the system.

Although much of our current effort is devoted to building a system for testing parallel matrix algorithms, we are also designing new parallel algorithms for important matrix processes. In particular we have developed a promising algorithm for the solution of the non-Hermitian eigenvalue problem. The method is based on a Jacobi-like iteration to reduce a matrix to upper triangular form by unitary transformations. It is numerically stable and parallelizes readily. Preliminary experiments indicate that it will be effective for a wide class of eigenvalue problems.

Work has also been done on parallel algorithms for solving sparse matrix problems that have a sparsity structure corresponding to a grid of points connected to up to eight nearest neighbors. Such problems arise in discretization of elliptic partial differential equations, network problems, and image processing. Various three-colorings of the graph and corresponding numberings of mesh points have been devised so that an iteration of a relaxation algorithm such as Gauss-Seidel or SOR can be executed with Parallelism comparable to the Jacobi algorithm, without degradation of convergence rate.

We have also been investigating algorithms for determining the equilibrium vector of nearly uncoupled Markov chains. These chains arise naturally in the stochastic modeling of computer systems. We have analysed the properties of a highly parallelizable method based on a combination of aggregation and the block Gauss-Seidel method.

Finally, we have coded a test package for the floating point processors, so that they may be quickly incorporated into the individual ZMOB boards.

Although at this time the project is still in a developmental state, we have given several talks on our work and have prepared papers for publication. These are listed in Appendix A.

Appendix A

I. Technical Reports

(1) G. W. Stewart, *Computing the CS Decomposition of a Partitioned Orthonormal Matrix*, TR-1159, May, 1982.

This paper describes an algorithm for simultaneously diagonalizing by orthogonal transformation the blocks of a partitioned matrix having orthonormal columns.

(2) G. W. Stewart *A Note on Complex Division*, TR-1206, August, 1982.

An algorithm (Smith, 1962) for computing the quotient of two complex numbers is modified to make it more robust in the presence of underflows.

(3) D. P. O'Leary, *Solving Sparse Matrix Problems on Parallel Computers*, TR-1234, December, 1982.

This paper has a dual character. The first part is a survey of some issues and ideas for sparse matrix computation on parallel processing machines. In the second part, some new results are presented concerning efficient parallel iterative algorithms for solving mesh problems which arise in network problems, image processing, and discretization of partial differential equations.

(4) G. W. Stewart, *A Jacobi-like Algorithm for Computing the Schur Decomposition of a Non-Hermitian Matrix*, TR-1321, August, 1983.

This paper describes an iterative method for reducing a general matrix to upper triangular form by unitary similarity transformations. The method is similar to Jacobi's method for the symmetric eigenvalue problem in that it uses plane rotations to annihilate off-diagonal elements, and when the matrix is Hermitian it reduces to a variant of Jacobi's method. Although the method cannot compete with the QR algorithm in serial implementation, it admits of a parallel implementation in which a double sweep of the matrix can be done in time proportional to the order of the matrix.

II. Technical reports in preparation

(1) D. P. O'Leary and G. W. Stewart, *Data Flow Algorithms for Matrix Computations,*

(2) G. W. Stewart and R. van de Geijn, *VMOB: Virtual ZMOB,*

(3) D. McAllister, G. W. Stewart, and W. J. Stewart, *A Two-Stage Algorithm for Nearly Uncoupled Markov Chains,*

(4) D. P. O'Leary, *Block Preconditionings for Parallel Computations,*

III. Presentations during 1983

(1) D. P. O'Leary, *Solving Mesh Problems on Parallel Computers,*
Bell Laboratory, Murray Hill, N.J., January, 1983
IBM T. J. Watson Laboratory, Yorktown Heights, N.Y., January, 1983.

(2) G. W. Sewart, *A Jacobi-like Algorithm for Computing the Schur Decomposition of a Non-Hermitian Matrix* (invited), Symposium on Numerical Analysis and Computational Complex Analysis, Zurich, Switzerland, August, 1983.

(3) G. W. Stewart, *The Structure of Nearly Uncoupled Markov Chains* (invited), International Workshop on Systems Modeling, Pisa, Italy, September, 1983.

(4) G. W. Stewart, *Data Flow Algorithms for Parallel Matrix Computations* (invited), SIAM Conference on Parallel Processing for Scientific Computing, Norfolk, VA, November, 1983.

(5) D. P. O'Leary, *Parallel Computations for Sparse Linear Systems* (minisymposium invitation), SIAM 1983 Fall Meeting, Norfolk, VA, November, 1983.

(6) D. C. Fisher, *Numerical Computations on Multiprocessors with Only Local Communications* (poster session), SIAM Conference on Parallel Processing for Scientific Computing, Norfolk, VA, November, 1983.

END

FILMED

11-85

DTIC